Web Services Technology Evaluation

ICNS - May 2, 2007

Authors: Steve Bliesner, Kelly Carter, Paul Comitz,
Avinash Pinto, Rich Schlegelmilch, Larry Shatos

- **Purpose**

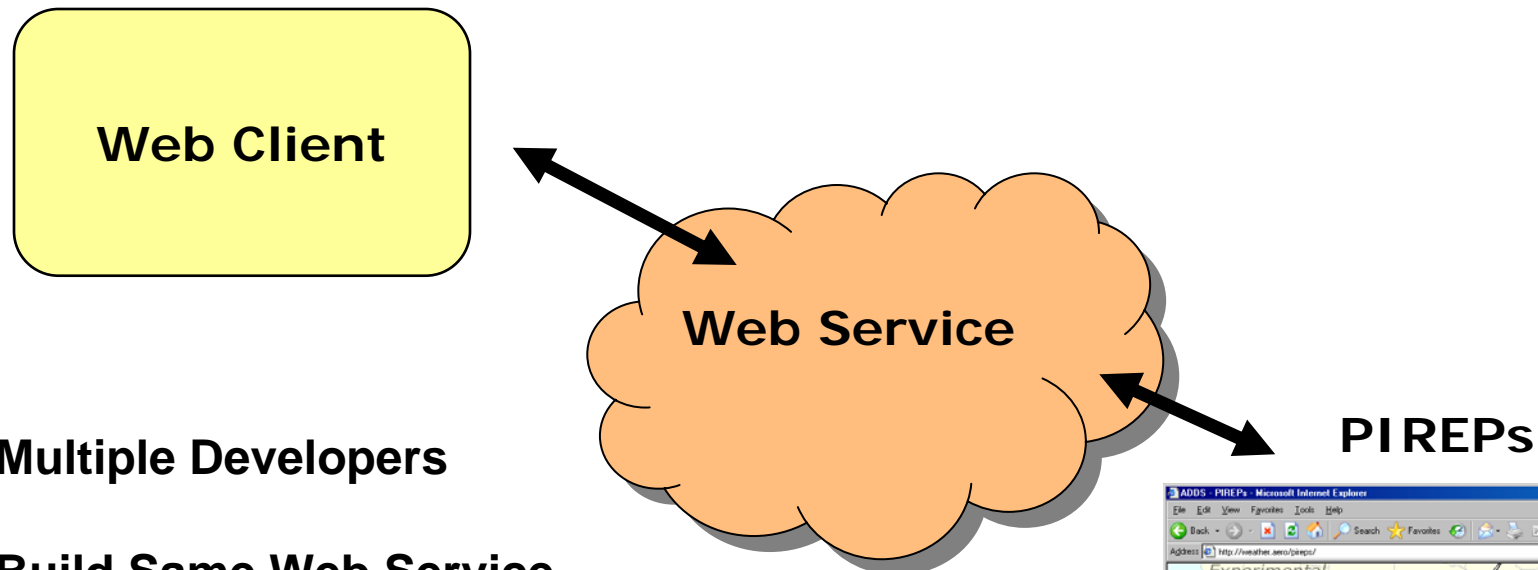- **Approach**

- **Comparisons**

- **Observations**

- **Evaluate Web Services technology offerings from multiple vendors**

- **Developer Assessments of :**
  - **Documentation**
  - **Support**
  - **Costs**
  - **Ease of Setup**
  - **Platforms**
  - **IDE**
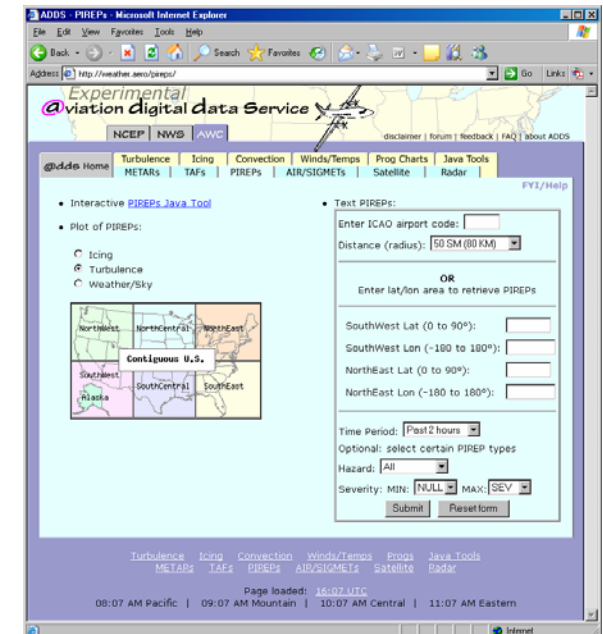 - **Use of standards**
  - **Scalability**

Web Client

Web Service

PIREPs

- **Multiple Developers**

- **Build Same Web Service**
  - **publish PIREPS**

- **Each developer uses a different vendor**
  - **.NET**
  - **BEA WLS**
  - **IBM Websphere Application Server**

- **Document development process**

Developer Notebooks

Blogs

- **Detailed logs of daily development activities**

- **Evaluation of criteria used to assess Web Service technology (documentation, cost, standards, etc)**

Result: Approximately 120 pages of working level notes

SOAP Request

Web Client

Interface (WSDL)

.NET

Weblogic

Websphere

SOAP Response
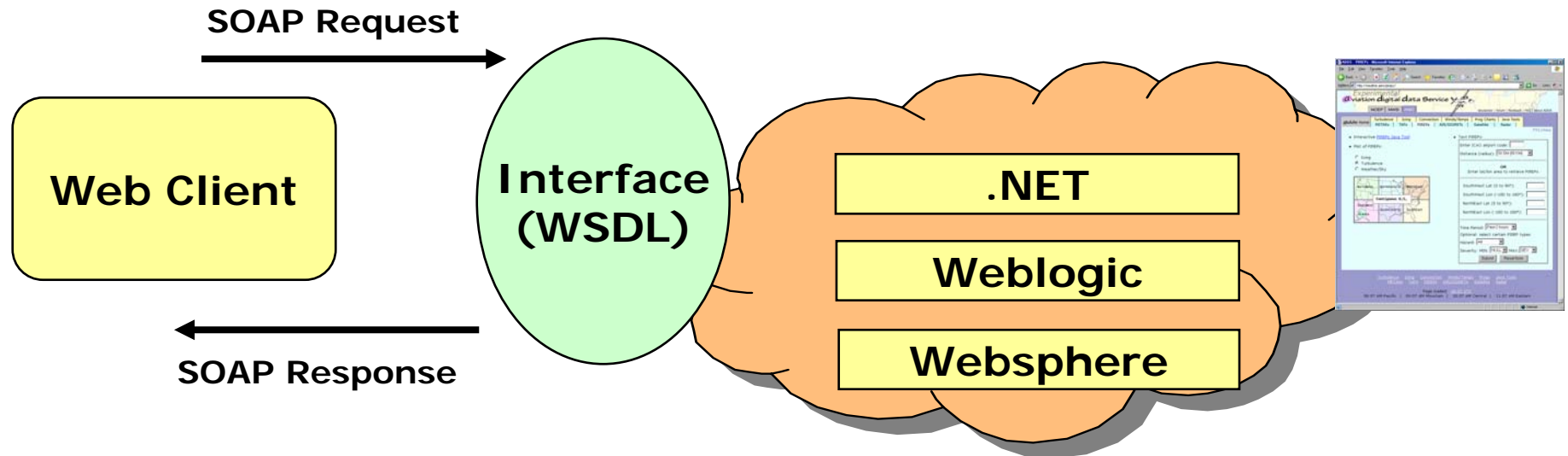
- Using WSDL, specify a common interface for client to Web Service interaction
- Develop identical Web Services using the candidate technologies

Web Services Technology Evaluation Comparison - .NET

Documentation	Thorough. Online and Offline. Third party training offered.
Support	Several options. Knowledgebase, FAQs, downloads. Free (MSDN) and paid support and training. Web forums.
Costs	.NET framework free – included in Windows. Microsoft Visual Studio costs. Hardware costs - $.
Ease of Setup	Straightforward – standard installation package. Updates available on website.
Platforms	.NET: many to all Microsoft operating systems on x86 architecture. MONO supports Linux, Mac OSX, Solaris, BSD, Windows, HP-UX on x86 architecture.
IDE	MS Visual Studio 2005 - $.
Standards	Supports many/most mainstream standards HTTP, SOAP, UDDI, XML, XSLT, DOM, SAX, etc.
Scalability	Requires further investigation.

Documentation	Documentation adequate. Sometimes referred to previous product version. Examples too trivial.
Support	Web forum available to ask and answer questions – extensively used by community.
Costs	Free SDK for development. Limited license for testing. License costs based on negotiated costs with vendor – vary by CPU - $. Hardware costs - $.
Ease of Setup	Easy. Installed from a web download. Ran a development server with a test configuration for web services.
Platforms	OS: Windows, Linux, HP-UX, Solaris, AIX. Architectures: x86, PA-RISC, Itanium, POWER4/5, AMD64, EM64T, Sparc.
IDE	NetBeans – free.
Standards	Supports many/most mainstream standards HTTP, SOAP, UDDI, XML, XSLT, DOM, SAX, etc.
Scalability	Requires further investigation.

Web Services Technology Evaluation Comparison – Websphere 6.1

Documentation	Documentation is adequate in some cases. Sometimes referred to previous product version. Extensive information in Red Books – sometimes there is so much information that is difficult to find what one needs
Support	IBM website can be hard to navigate at times. Typically support was found via Google by entering the error code. Free online support available with download of trail application server.
Costs	Free trial application server available. License costs based on negotiated costs with vendor – vary by CPU - $. Hardware costs - $. The cost of professional services after the purchase should be included as part of the cost,
Ease of Setup	Easy. Installed from a web download. Updates available on line. Some installers and uninstallers do not work.
Platforms	OS: AIX, HP-UX, Z/OS, Linux, Solaris, Windows
IDE	NetBeans, Eclipse – free. Rational Application Developer - $
Standards	Supports many/most mainstream standards HTTP, SOAP, UDDI, XML, XSLT, DOM, SAX, etc.
Scalability	Requires further investigation.

- **Microsoft .NET, BEA Weblogic, and IBM WebSphere Web Services technologies provide sufficient capabilities necessary for data access and distribution**

- **Services were developed and tested inexpensively over the course of 6 weeks by experienced developers**

- **Services did not address time criticality or high bandwidth issues**

- **Deploying a MONO web service with .NET code was surprisingly simple**

- **More expensive products do not always have better documentations**

- **Experience:**
  - **Some product cannot be used without engaging vendors professional services organization**
    - **Very Expen$ive**

- **Some unsupported open source products are easier to use than vendors supported products**
  - **If open sources product is hard to use or does not work per documentation, developers quickly move on**

  - **Significant sunk costs requires a commitment to the specific vendor technology**
    - **No choice but do engage expensive technology services**

> **Are some open source products better because they MUST be?**

- **Will use of Web Services save money?**
  - **There is no evidence of this**

- **Will Web Services save time ?**
  - **There is no evidence of this**

- **Are Web Services appropriate for architectures that are built on a "portfolio of network addressable services"**
  - **Yes – but**
    - **We believe that additional domain specific engineering is required**
      - meta data strategies
      - abstract messaging interfaces
      - More emphasis on a domain specific information layer

- **Other considerations**
  - **Pub/Sub support is still emerging**
  - **Web Services are not appropriate for real time**
  - **Web Services are exploding in complexity**
    - **Wasn't this the motivation for Web Services in the first place – Simple Object Access Protocol**